Implementing Trust-Based Decision Making for Health IoT Systems Using Block chain

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology

in

Computer Science

by

AKSHAT BHANCHAWAT

17BCE0195

CHINMAY KALVADE

17BCE0912

SARTHAK SURANA

17BCE0996

DHRUV MITTAL

17BCE2110

Under the guidance of

Prof. / Dr. Krishnamoorthy A.

SCOPE

VIT, Vellore.

June, 2020

Abstract

Among the panoply of uses empowered by the Internet of Things (IoT), keen and associated human services is an especially significant one. This paper displays a Blockchain based health IoT system. The proposed system can control IoT devices to utilize the most dependable natural health data for basic leadership for the sake of the device client. Utilizing blockchain for IoT information gives a solid and a proficient conveyed trust-based basic leadership system.The next section introduces the trust-based concept and the need of blockchain in the existing system.

Keywords— Microprocessor, Raspberry Pi, Internet of things (IoT), trust management, decision trust, health IoT, trust-based decision making, blockchain

CONTENTS

1. INTRODUCTION

1.1 OBJECTIVE

This paper presents a Blockchain based health IoT system which provides a reliable and an efficient distributed trust-based decision-making system. A health IoT system comprises of Smart IoT devices which offer location based data by utilizing individual zone systems. A remote territory system can likewise be incorporated to get progressively dependable data. The sole point of such IoT system is to settle on dependable choices for the benefit of its clients with the end goal that their health isn't undermined. Information Integrity is the key in IoT applications. The trustfulness of information straightforwardly influences the basic leadership system. Beforehand many trust-based conventions have been executed in different IoT fields including medicinal services. Unlike previous implementations of IoT in healthcare (base paper [1]), our paper focuses on secured implementation of the existing protocols.

1.2 MOTIVATION

According to Cisco [10], “the IoT will comprise more than 30 billion connected devices". With such a significant number of associated devices confiding in an incorporated server, makes all the IoT devices powerless against assaults and dangers on the double and hampers the uprightness of IoT system. One of the significant restrictions of IoT, as we probably am aware it today, is that it depends on concentrated, handled correspondence models. This is referred to in like manner speech as the server/customer correspondence model and most conventional innovation depends on it.

In this manner, to address this issue, Thus, to address this issue, we propose a decentralised system using blockchain provides maximum security by eliminating single points of failure. This would help make consumer data more private. Enhancement compared to existing models:

Dissimilar to existing trust-based conventions ([1], [4] and [5]), where IOT model doesn't consider situations where malignant hubs harm the good hubs to demolish their trust score by utilizing sassing assaults, and furthermore utilizing polling form stuffing assaults toward one another to help their trust scores. Such assaults are just conceivable in situations where

the hubs are tempered by outside means and made to act predisposition. Blockchain takes out the issue by giving a decentralized system which makes device availability and information stockpiling trust less through hubs that can work without a brought together power.
Besides, having a distributed correspondence model rather than the standard server/customer one can be the manageable arrangement the IoT business is searching for. Distributed correspondence will lessen the expenses of introducing and keeping up costly servers. Calculation and capacity needs will be conveyed crosswise over a huge number of IoT devices and no focal disappointment will bring about disappointment of the entire system. The decentralization of the blockchain record guarantees that calculation and capacity are spread crosswise over a huge number of devices and not on one focal server. Accordingly, the circumstance where server disappointment brings about a breakdown of the whole IoT system will never again exist. When blockchain and IoT meet up, keen devices will have the option to trade information and even attempt money related exchanges through a decentralized, trust less blockchain. Accordingly, there will never again be any reliance on an incorporated position. Decentralized system likewise settles many existing issues in brought together design.

1.3 BACKGROUND

The base paper proposes a trustbased decision making system which considers [1] classification of risk, reliable trust, and loss of health probability as key parameters for decision making.The proposed model gathers information from sensors and ascertain the any wellbeing misfortune likelihood. In view of client's defenselessness the framework settles on a choice is the inquiry area ought to be visited by client or not thinking about his/her medical problems.
Trust assessment is also used for monitoring the health data stored in cloud to ensure more secure access control [2].A Trust Authority is set up which relegates an authentication to every client dependent on the trust level. This testament is utilized to check the client while the client attempts to refresh/change the data put away in the e-cloud.
There are IoT frameworks in which trust appraisal is finished by investigating the information of the earth gathered by the individuals from a network [3]. The gadgets utilize the adaptability of the framework to get trust evaluations which makes the framework progressively solid. The objective of the framework is to give the individuals from the network data which can be trusted and choices can made upon it.
A social Internet of Things (IoT) [4]can be viewed as a blend of conventional P2P systems

and informal communities in which IoT gadgets connect with and set up associations with each extraordinary to accomplish a typical point. The paper considers a client focused social IoT condition without a brought together confided in power. Every gadget has its one of a kind personality and hubs having a place with an equivalent arrangement of gatherings are probably going to have comparable objectives.

Trust and Reputation Management is additionally utilized for remote sensor systems [5] which are inclined to security dangers on account of their remote and plan nature. WSN hubs are obliged gadgets which require memory and vitality effective calculations which additionally don't bargain the security. The paper proposes a trust-based administration conspire which uses surveying of trust votes in favor of basic leadership to handle the potential dangers looked by the framework.

Body Location Network likewise use trust evaluation techniques [6]. A body area system is remote arrange gathering of sensors, wearable and implantable gadgets that put in or around the collection of patient for checking and mechanizing, transmitting body information. Boycott hub trade touchy and significant therapeutic information among them self or with cloud servers. Boycott have as of late addition consideration on security of this. A large portion of the BAN security arrangement are takes a shot at one of encryption, get to control, key trade, confirmation. These are great strategies however these are tedious and required more calculation. In the featured paper, new extraordinary methodology which is proposal framework. It is basic to confide in a gadget which isn't recently associated and trading the information first time. In such situation, gadget get proposal from other gadget which is a piece of a similar BAN organize. Furthermore, as indicated by suggestion, gadget will trade information with new gadget.

Trust Chain [7], proposes Blockchain, as a decentralized framework, which disposes of a confided in outsider by empowering individuals to affirm data exactness and assurance its lastingness. IoT gadgets can use blockchain to enroll themselves and sort out, store, and offer floods of data suitably and reliably.

The paper[8], examines the properties of trust, propose targets of IoT trust the board, and supply a study on the present writing propels towards reliable IoT. in addition, we talk about unsolved issues, determine research difficulties and show future investigation slants by proposing a model for all encompassing trust the board in IoT.

Additionally investigated [9]the trust properties that effect trust connections, grouped them into 5 classes and demonstrated that all encompassing trust the executives should concern half or every one of them in a few settings and for different capacities. upheld a general IoT

framework model, we arranged goals for all encompassing IoT trust the board and showed their supporting IoT layers by activity vertical trust the board is urgent for accomplishing dependable IoT.

The paper[10] investigations give a security examination of patient observing for the IoT wellbeing's essential security issues which will put the eHealth framework in peril. the specific security objectives and necessities, vulnerabilities, dangers, and assaults area unit dissected and a couple of potential security proposals with course for future work area unit referenced. It extra talks about why security issues need flexible, setting mindful, and adaptational security components. while making a security choice wellbeing component should consolidate security needs, dangers, and assaults bolstered the patient's area and natural setting.

Different trust-based decision-making systems are compared based on some objectives which should be accomplished by a trust-based health IoT system.

Table 1: Literature Survey Table

Papers	**Parameters**			
	Trust relation and decision	**Identity trust**	**System security and robustness**	**Data perception trust**
[1]	Yes	Yes(Uses feedback mechanism)	Partial	Yes
[2]	Yes	Yes (certificate security)	Yes	No
[3]	Yes	No	Partial	Yes (Cluster based data collection)
[4]	No	No	No	Yes(based on social IOT)
[5]	Yes	Yes (Uses polling scheme)	No	Yes
[6]	Yes	Yes (based on peer recommendation)	Yes	No

[7]	Yes	Yes	Yes	No
[8]	Yes	No	Yes	Yes
[9]	Yes	Yes (Access Control Sceme)	Yes	No

2. PROJECT DESCRIPTION AND GOALS

Our proposed IoT system architecture differs from the existing centralised architecture.

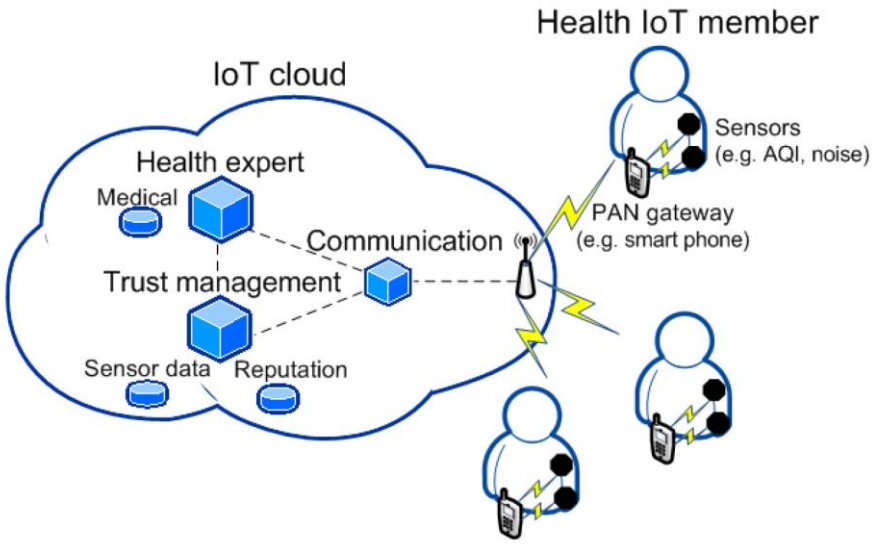


Figure 1: Existing Cloud Architecture

We arranged goals for all encompassing IoT trust the board and showed their supporting IoT layers by activity vertical trust the board is urgent for accomplishing dependable IoT.

Our Proposed Decentralised health IoT Architecture:

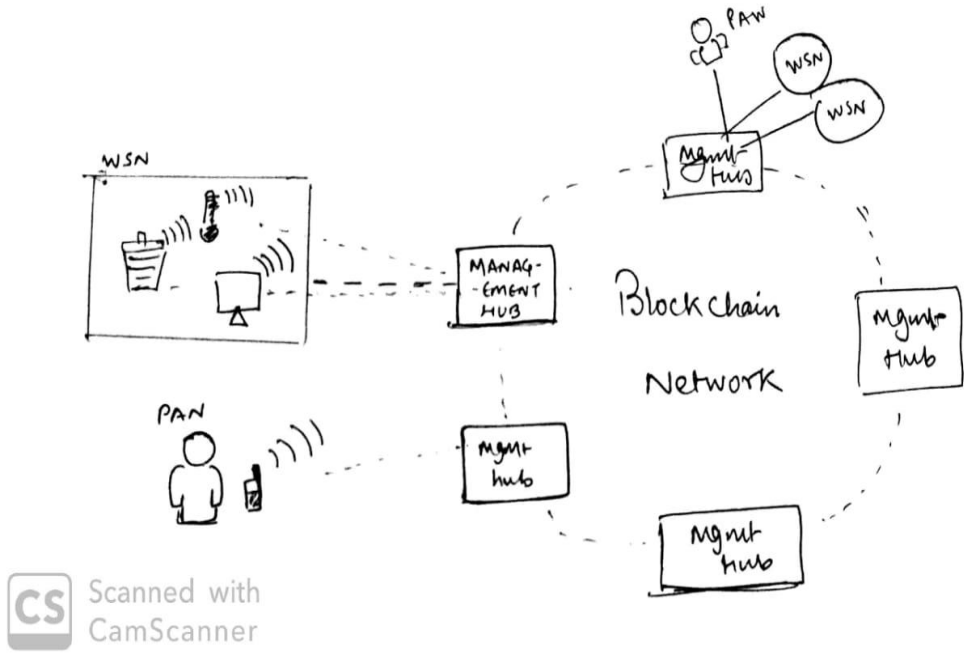


Figure 2: Decentralized Architecture

3. TECHNICAL SPECIFICATIONS

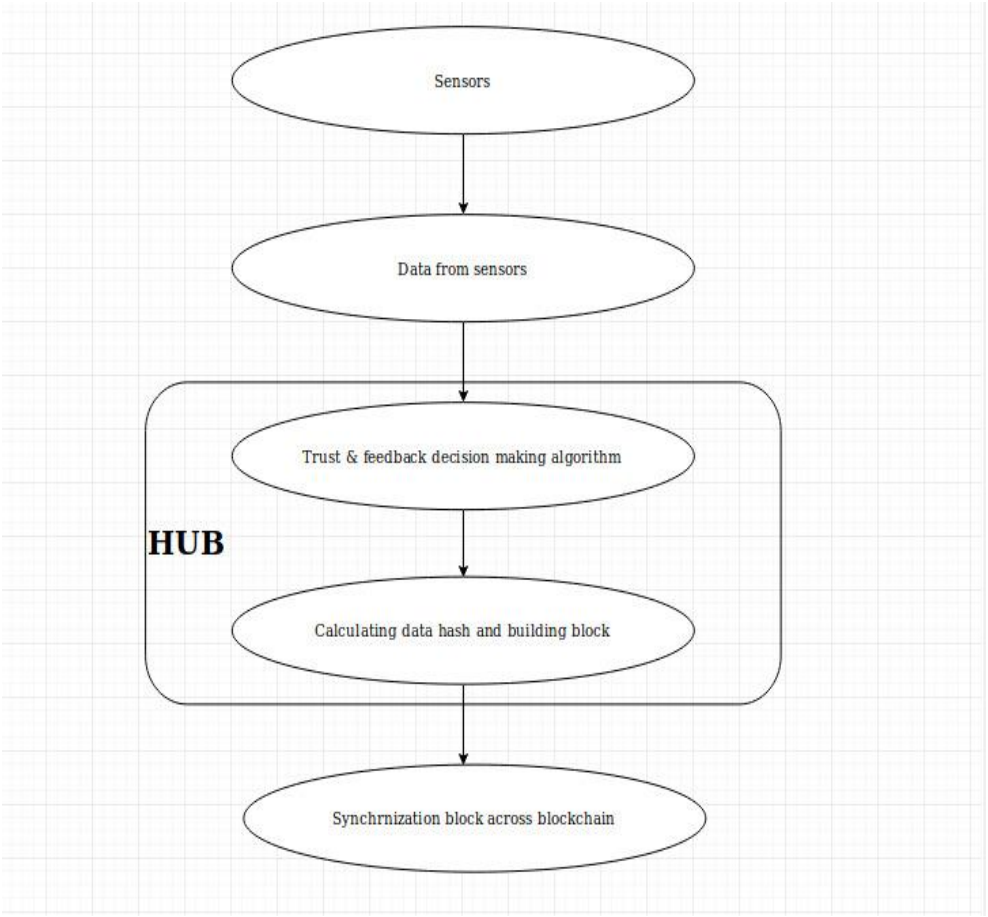


Figure 3. Flow chart

3.1 Decentralised Architecture: Blockchain

Data from various PAN and WSN networks would be passed to the nearest management hub. The authorised can add a block to the chain once it has been approved by rest of the blocks in the chain.

Since data of each block is shared among all the other blocks, so if some unauthorised tries to change the data, the hash value of the block is also changed and thus the successive block which contains the hash value of previous block will have a mismatch in hash with the previous block. A change in a single block will make all the following blocks invalid.

Blockchain uses hashing and proof of work as a security feature.

Unlike cloud architecture, in a decentralised system each management hub has been accustomed with decision making algorithm. The **management hub** the most important part of the system is basically a computational device which is responsible for:

1. Collection of data from the WSNs or PANs or individual sensors

(Query / Response Module).

2. Taking nearest user's queries and giving decisions for the query as responses.

(Decision Making Module)

3. Calculating the hash and adding block to the existing blockchain.

(Mining Module)

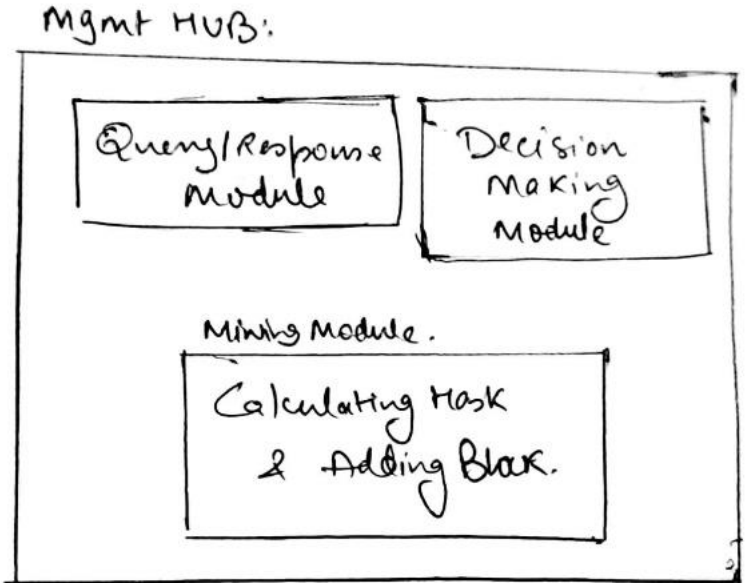


Figure 4. MANAGEMENT HUB

3.2 Query/Response Module:

This module takes care of the communication of Management hub with the users as well as with the ground sensors.

Data from Sensors: Data fetched from each ground and location sensor per 3 minutes is formatted, structured properly and sent to the mining module where the sensors data is added in the blockchain.

Users Queries: It processes the user's queries via user interface, related to a particular location about their health and sends the data further to decision making module for decision making. The resulting decision is displayed to the user.

Feedback from User: Feedback is also collected from the user related to decision making and any feedback rating from user affects the trust rating of the sensors. If the feedback is positive the rating is increased else it is decreased accordingly. The feedback data is sent to the mining module which adds this data as a block in the existing blockchain.

3.3 Decision Making Module:

Each management hubalso comprises of a decision-making module.The decision-making module takes in user's queries for e.g. If a user wants to visit location x, then he/she enquires if the queried location is fit for them or not.

When decision module is called, it fetches the latest added block in the blockchain , to get the latest updates related to that particular location and following the decision making algorithm which basically works on thresholding , a decision is passed back to the query/response module.

According to decision making algorithm, each user is assigned a vulnerability index (for each user vulnerability index can correspond to different phenomenon) and if the calculated value is greater than threshold vulnerable index then the system would not allow user to visit the particular location.

The module uses following parameters for decision making such as:

1.Decay factor (i.e. Data Timestamp)

2.Capability of Devices

3. Location Rating

4. Trust-Rating of Device

5. Location experience

Location Rating: Using the data from sensors each location is rated for a particular phenomenon.

Trust–Rating Module: The management hub periodically calculates every member device's trust score. If the trust score of a particular device is less than the threshold trust, then the particular device/node is declared as malicious and removed from the system.

Trust depends on three factors:

1.Feedback from the query user after he/she visits the location. If a positive feedback then, the existing WSNs can be trusted.

2.keeping a check on the feedback user if his/her feedback is true or not, by taking aggregate of several feedbacks.

3. Different members can also keep a check on other users if they were present at a particular location during the rating, since they were able to communicate over short range transmission (functioning as a substitution for physical eye sight).

All these factors decide the rating of the existing devices for a particular phenomenon.

3.4 Mining Module:

Whenever data is received from the query module, the hash for that data is calculated and added as a block to the existing blockchain. The mined block contains the latest sensed value from the sensor, along with its trust rating.

Thus, leveraging the blockchain technology in implementation of trust-based protocol presents a robust decision-making health IoT system.

4. DESIGN APPROACH AND DETAILS

4.1 Design Approach/ Materials and Methods

Materials Used: Raspberry Pi 4b, Node MCU, Power bank, LEDs, Jumper cables, USB cables.

Sensors used: Smoke Sensor and Humidity sensor

For the implementation, we took 4 management hubs as a part of our blockchain based IoT system. Raspberry pi 4b were deployed as management hubs and each raspberry pi is

connected to some ground sensors (PAN/WANs) and a user interface.

A dedicated User Interface where user can query related to a particular location using his login credentials. The management hub provides the user with the decision and also asks for their feedback. The user can give a star rating feedback, based on which the sensors trust rating is altered. Also, for testing purpose we took testcases for two locations namely x and y, for which different users can query and get results related to their health phenomenon.

4.2 Codes and Standards

Basic Block of a blockchain implemented using NODEjs:

```
const ChainUtil = require('../chain-util');
const { DIFFICULTY, MINE_RATE } = require('../config');

class Block
{
    constructor(timestamp, lastHash, hash, data, nonce, difficulty){
        this.timestamp = timestamp;
        this.lastHash = lastHash;
        this.hash = hash;
        this.data = data;
        this.nonce = nonce;
        this.difficulty = difficulty || DIFFICULTY;
    }
    toString(){
        return `Block -
        Time Stamp : ${this.timestamp}
        Last Hash  : ${this.lastHash.substring(0, 10)}
        Hash       : ${this.hash.substring(0, 10)}
        Nonce      : ${this.nonce}
        Data       : ${this.data}
        Difficulty : ${this.difficulty}
        `;
    }

    static genesis() {
        return new this("Genesis Block", '------', 'f1r5t block', [], 0,
DIFFICULTY);
    }

    static mineBlock(lastBlock, data) {
        const lastHash = lastBlock.hash;
        let nonce = 0;
        let { difficulty } = lastBlock;
```

```
        let timestamp, hash;
        do {
            nonce++;
            timestamp = Date.now();
            difficulty = Block.adjustDifficulty(lastBlock, timestamp);
            hash = Block.hash(timestamp, lastHash, data, nonce, difficulty);
        } while (hash.substr(0, difficulty) !== '0'.repeat(difficulty));
        return new this(timestamp, lastHash, hash, data, nonce, difficulty);
    }

    static hash(timestamp, lastHash, data, nonce, difficulty) {
        return
ChainUtil.hash(`${timestamp}${lastHash}${data}${nonce}${difficulty}`).toString()
;
    }
    static genHash(block) {
        const {timestamp, data, lastHash, nonce, difficulty} = block;
        return  Block.hash(timestamp, lastHash, data, nonce, difficulty);
    }
    static adjustDifficulty(lastBlock, currentTime) {
        let { difficulty } = lastBlock;
        difficulty = lastBlock.timestamp + MINE_RATE > currentTime ?
difficulty+1 : difficulty-1;
        return difficulty;
    }
}
module.exports = Block;
```

Mining a block:

```
const Block = require("./block");
describe('block', ()=>{
    let block, lastBlock, data;

    beforeEach(()=>{
        data = 'Hello Unknown';
        lastBlock = Block.genesis();
        block = Block.mineBlock(lastBlock, data);
    });

    it('sets the `data` match to input data', ()=>{
        expect(block.data).toEqual(data);
    });
```

```
    it('sets the `lastHash` match to hash of last block hash', ()=>{
        expect(block.lastHash).toEqual(lastBlock.hash);
    });
    it('generate a hash that matches the difficulty', ()=>{
        expect(block.hash.substring(0,
block.difficulty)).toEqual('0'.repeat(block.difficulty));
    });
    it('lowering the difficulty level when mining slow', ()=>{
        expect(Block.adjustDifficulty(block,
block.timestamp+5600)).toEqual(block.difficulty-1);
    });
    it('raising the difficulty level when mining too fast', ()=>{
        expect(Block.adjustDifficulty(block,
block.timestamp+1)).toEqual(block.difficulty+1);
    });
});
```

Code for NODE MCU

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
//#include <ArduinoJson.h>
int redLed = D5;
int greenLed = D1;
int smoke  = A0 ;
// Your threshold value
int sensorThres = 690;
#define LED D0
const char *ssid =  "VIT2.4G";
void request_send(String *, int *);
void setup() {
  Serial.begin(115200);
  String thisBoard= ARDUINO_BOARD;
  Serial.println(thisBoard);
  Serial.println("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");

  pinMode(redLed, OUTPUT);
```

```
  pinMode(greenLed, OUTPUT);
  pinMode(smoke , INPUT);
  pinMode(LED, OUTPUT);
}
void loop() {
  int analogSensor = analogRead(smoke);
  digitalWrite(LED, HIGH);
  Serial.print("Pin A0: ");
  Serial.println(analogSensor);
  // Checks if it has reached the threshold value
  if (analogSensor > sensorThres)
  {
    Serial.println("Something happening abnormal");
    digitalWrite(redLed, HIGH);
    digitalWrite(greenLed, LOW);
  }
  else
  {
    Serial.println("Everything is alright");
    digitalWrite(redLed, LOW);
    digitalWrite(greenLed, HIGH);
  }
  //sending data to normal pc
  request_send("http://192.168.43.151:3001/smoke-data/", analogSensor);
  //sending data to rpi
  //request_send("http://192.168.43.124:3001/smoke-data/", analogSensor);
  delay(1000);
  digitalWrite(LED, LOW);
  delay(10000);
}
void request_send(String url, int data){

  HTTPClient http;  //Declare an object of class HTTPClient
  http.begin(url);  //Specify request destination

  http.addHeader("User-Agent","Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv","66.0) Gecko/20100101 Firefox/66.0");

http.addHeader("Accept","text/html,application/xhtml+xml,application/xml;q=0.9,/
;q=0.8");
  http.addHeader("Accept-Language","en-US,en;q=0.5");
  http.addHeader("Accept-Encoding","gzip, deflate, br");
  http.addHeader("DNT","1");
  http.addHeader("Connection","keep-alive");
  http.addHeader("Upgrade-Insecure-Requests","1");
  http.addHeader("Cache-Control","max-age=0");
```

```
  http.addHeader("Content-Type", "application/x-www-form-urlencoded");
  String postData = "data="+String(data);
  int httpCode = http.POST(postData);
  String payload = http.getString();   //Get the request response payload
  Serial.println(payload);
  http.end();
}
```

Peer to peer (P2P) Server

```
const WebSocket = require('ws');
const P2P_PORT = process.env.P2P_PORT || 5001;
const peers = process.env.PEERS ? process.env.PEERS.split(',') : [];
const MESSAGE_TYPE = {
    chain: 'CHAIN',
    feedback: 'FEEDBACK'
};
class P2pServer {
    constructor(blockchain, feedback) {
        this.blockchain = blockchain;
        this.feedback = feedback;
        this.sockets = [];
    }
    listen() {
        const server = new WebSocket.Server({ port: P2P_PORT });
        server.on('connection', socket => this.connectSocket(socket));
        this.connectToPeer();
        console.log(`Listen for P2P connection on port ${P2P_PORT}`);
    }
    connectToPeer() {
        peers.forEach(peer =>{
            const socket = new WebSocket(peer);
            socket.on('open', () => this.connectSocket(socket));
        });
    }
    connectSocket(socket) {
        this.sockets.push(socket);
        console.log('Connection established');
        this.messageHandler(socket);
        this.sendChain(socket);
    }
    sendChain(socket) {
        socket.send(JSON.stringify({ type: MESSAGE_TYPE.chain, chain:
this.blockchain.chain }));
    }
    sendFeedback(socket, feedback) {
```

```
            socket.send(JSON.stringify({ type: MESSAGE_TYPE.feedback, feedback }));
        }
        messageHandler(socket) {
            socket.on('message', message => {
                const data = JSON.parse(message);
                switch (data.type) {
                    case MESSAGE_TYPE.chain:
                        this.blockchain.replaceChain(data.chain);
                        break;
                    case MESSAGE_TYPE.feedback:
                        this.feedback.replaceData(data.feedback);
                        break;
                }
            });
        }
        broadcastFeedback(feedback) {
            this.sockets.forEach(socket => {
                this.sendFeedback(socket, feedback);
            });
        }
        syncChain() {
            this.sockets.forEach(socket => {
                this.sendChain(socket);
            });
        }
    }
    module.exports = P2pServer;
```

4.3 Constraints, Alternatives and Tradeoffs

Although the key idea of blockchain is simple, its implementation poses a great number of challenges. This section introduces the main ones that its use brings about.

Resource Constraints

General blockchain requires high computational power, high bandwidth, and low delays. Most blockchain systems use Proof-of-Work (PoW) as their underlying consensus mechanism. However, the mining process in PoW requires huge computational power. Most

IoT devices have simple hardware specifications and low processing power. It is not capable or time consuming for IoT devices to perform the mining tasks of blockchain.

Apart from this, blockchain needs to perform data encryption frequently. However, the encryption speed and time will be different, because different IoT devices have different computational power. Moreover, other processes, consistency algorithms, and routine testing require huge processing power, which overloads the low power capacity of IoT devices. Moreover, in blockchain, the transactions and blocks are not stored in a central server. But some nodes need to keep a copy of the full ledger in their storage. The size of the ledger will increase over time. However, the majority of IoT devices have low hardware storage capacity. Low power IoT devices only have 10KB to 100KB memory for storing data and memory. But blockchain requires massive storage for storing the entire chain. For example, Bitcoin needs over 200 GB of memory, Etherum requires over 1.5 T of memory. It is not capable of storing a copy of the full blockchain for IoT components.

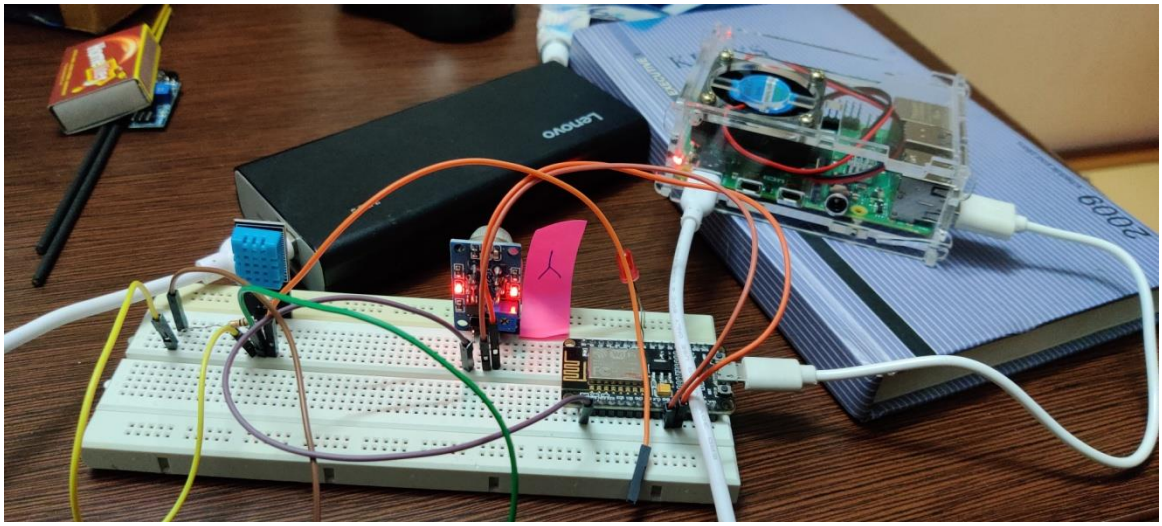
Additionally, the consensus process in blockchain requires the exchange of information between nodes frequently to reach an agreement to maintain the correctness of blockchain and generate new blocks. This process requires high bandwidth and low network latency. However, IoT devices are always strict in a limited bandwidth.

Storage capacity and scalability

Storage capacity and scalability have been deeply questioned in blockchain. In this technology, the chain is always growing, at a rate of 1MB per block every 10 min in Bitcoin, and there are copies stored among nodes in the network. Although only full nodes (a node that can fully validate transactions and blocks) store the full chain, storage requirements are significant. As the size grows, nodes require more and more resources, thus reducing the system’s capacity scale. In addition, an oversized chain has negative effects on performance, for instance, it increases synchronization time for new users.

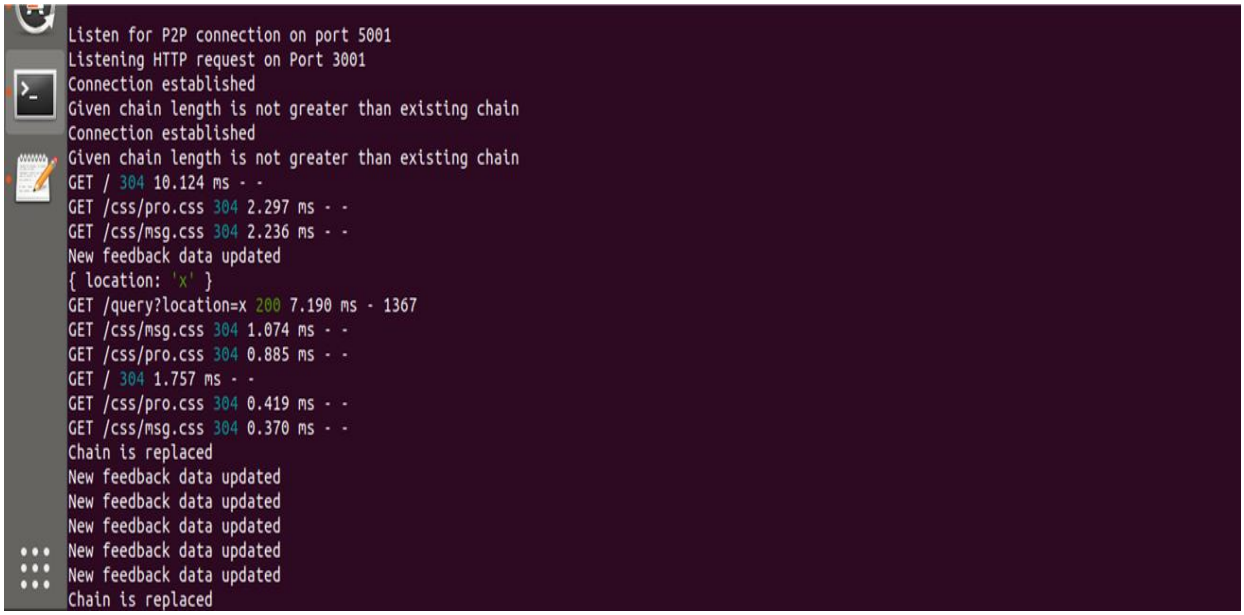
Transaction validation is a key component of the distributed consensus protocol as nodes in the blockchain network are expected to validate each transaction of each block. The number of transactions in a block and the time between blocks, modulate the computational power required and this has a direct effect on transaction confirmation times. Hence, the consensus protocol has a direct effect on the scalability of blockchain networks.

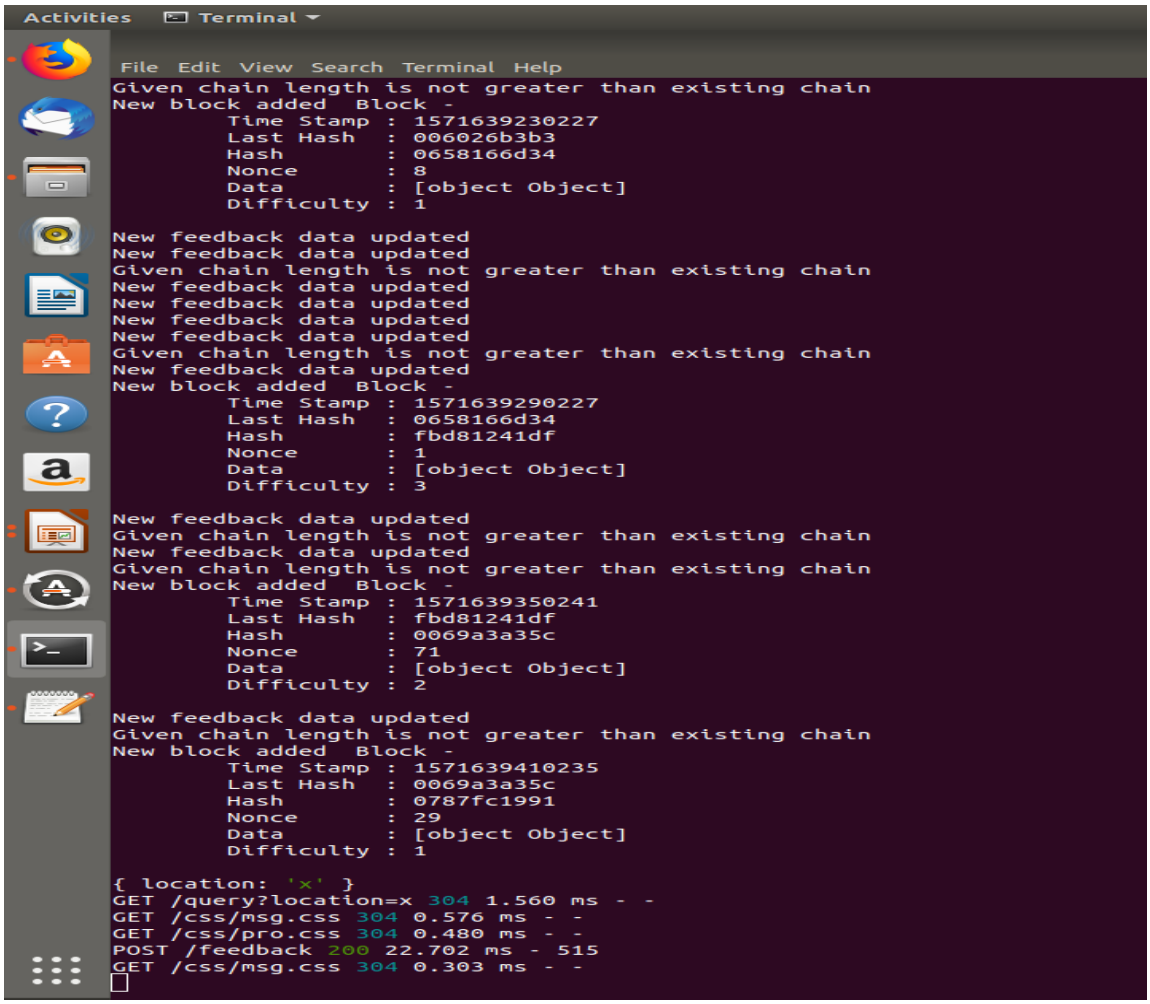
5. **PROJECT DEMONSTRATION**



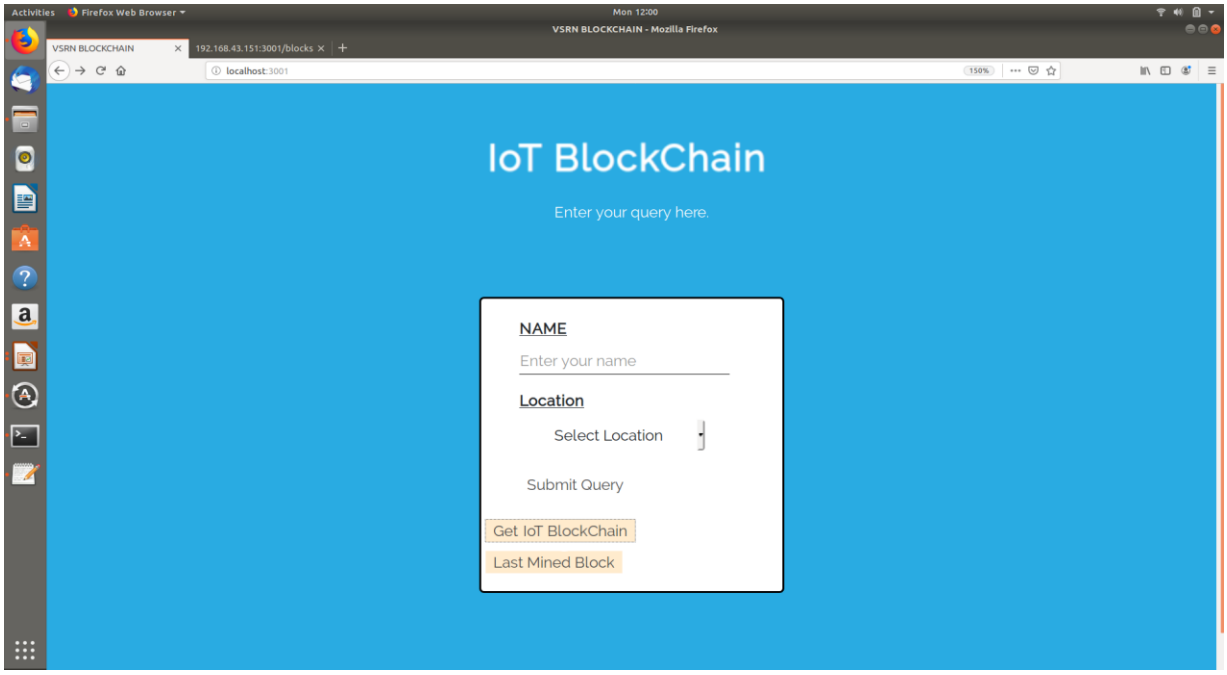
Connection of Raspberry pi(Management Hub) with Sensors

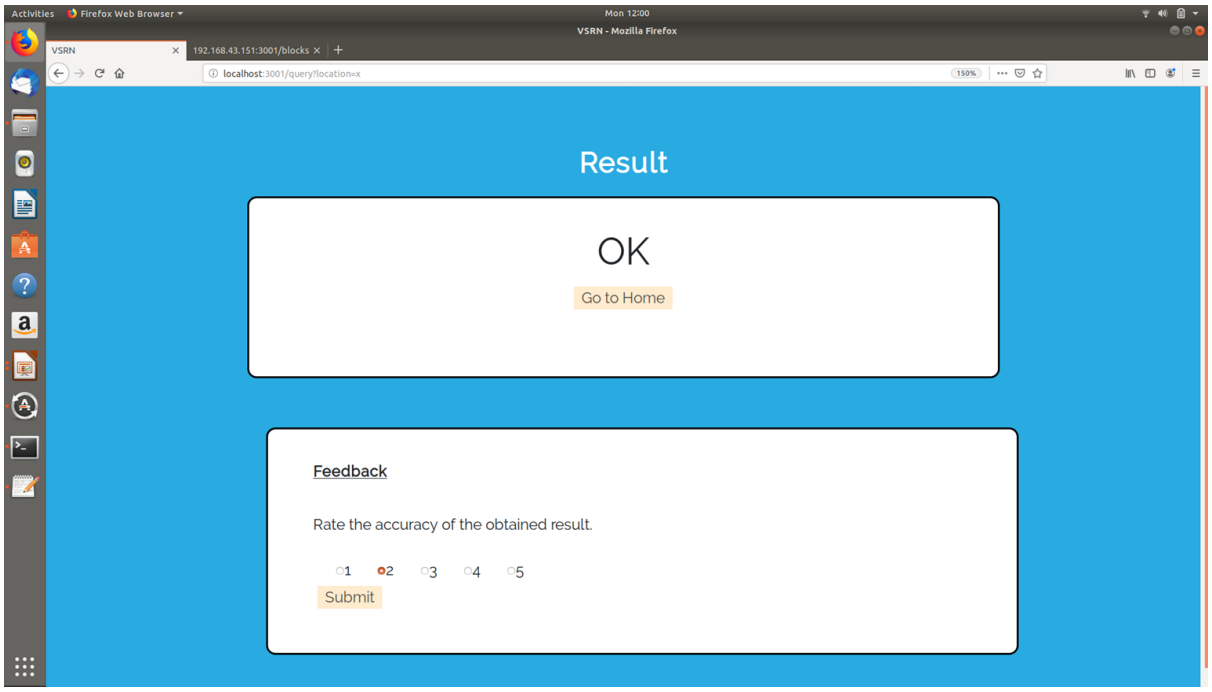
Blockchain Initiation:

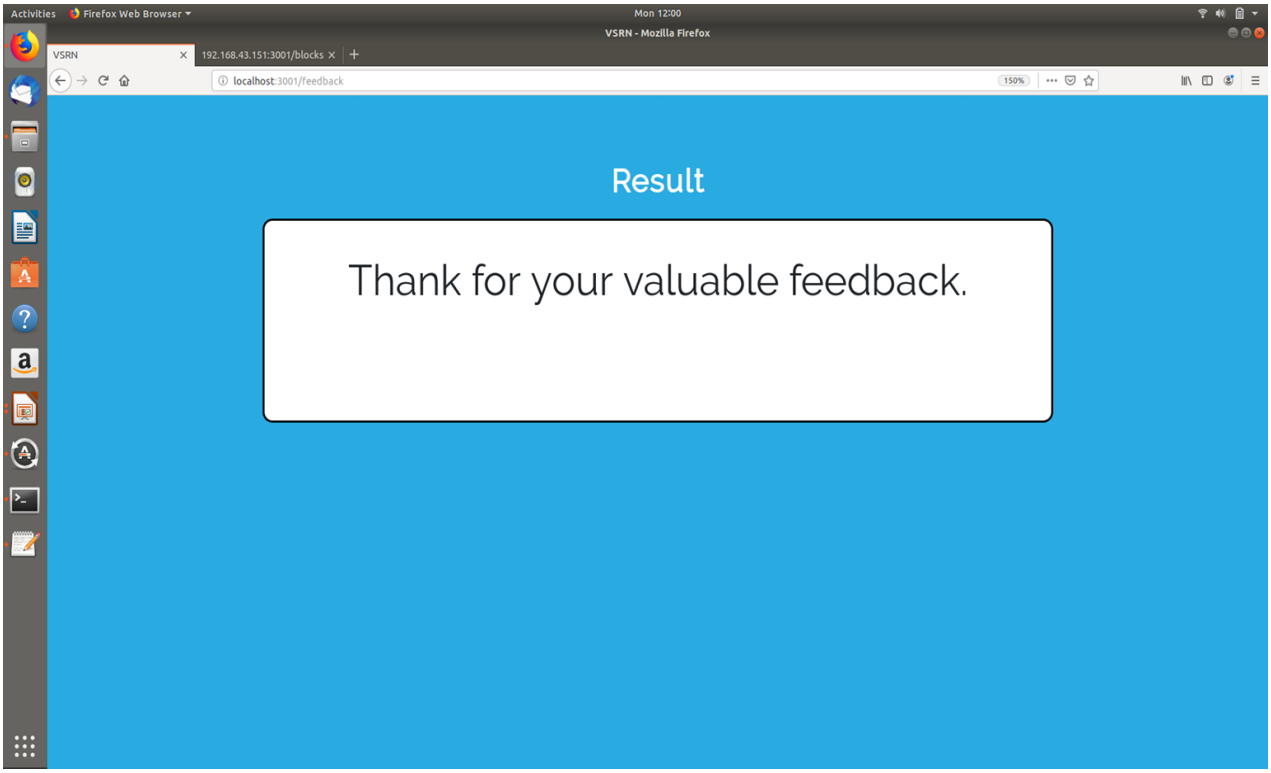




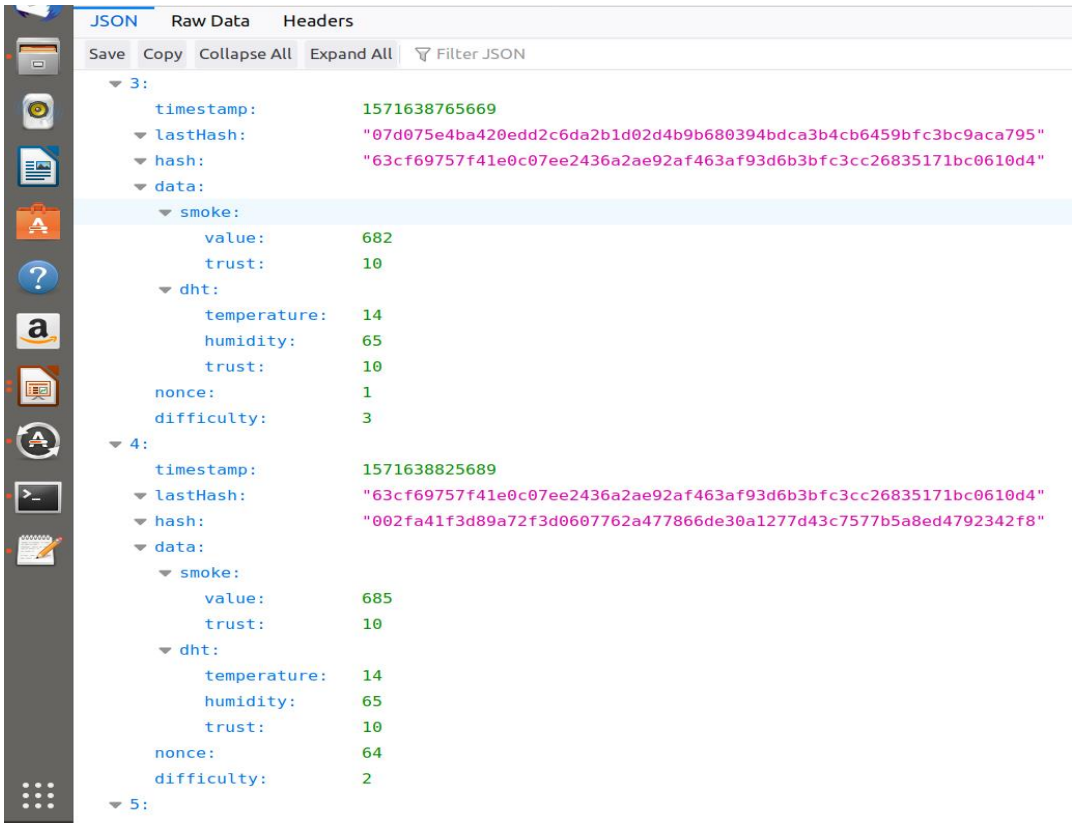
User Interface:




Activities
Firefox Web Browser
Mon 12:00
VSRN - Mozilla Firefox
VSRN
192.168.43.151:3001/blocks
localhost:3001/query?location=x
150%
Result
OK
Go to Home
Feedback
Rate the accuracy of the obtained result.
1
2
3
4
5
Submit


Activities
Firefox Web Browser
Mon 12:00
VSRN - Mozilla Firefox
VSRN
192.168.43.151:3001/blocks
localhost:3001/feedback
150%
Result
Thank for your valuable feedback.

Minining Data History:



6. RESULT & DISCUSSION

At the end of implementation of this project, a decentralized trust-based health IOT system using blockchain is implemented. The system is facilitated by a user interface which gives user the accessibility to query the system for health-related decisions. The blockchain mining module adds the data from sensor every minute to the blockchain. When the user queries, the query/response module is fetching the latest block to take decisions. The deployed system functions well for different testcases and makes right decisions. The feedback mechanism helps maintain the trust value of the sensors.

7. SUMMARY

This project is able to achieve its aim of implementing a much more secure trust-based health IOT system using Raspberry Pi microprocessors. The technology of IOT and blockchain both being distributive, complement each other very well to provide a robust and much more secure health IoT system compared to existing centralized architectures. The decentralized system not only takes care of the trust of the IoT devices but also the incorruptible system and helps maintain data integrity. The system with some future optimizations is fit to be implemented at large scale.

References

[1] Security Analysis of a Patient Monitoring System for the Internet of Things in eHealth , eTELEMED 2015 : The Seventh International Conference on eHealth, Telemedicine, and Social Medicine ,Kashif Habib, Arild Torjusen, Wolfgang Leister Norwegian Computing Center Oslo, Norway

[2] "Survey on trust management for Internet of Things, Zheng Yan, Peng Zhang, Athanasios V. Vasilakos"

[3] "Reputation-based Trust Management in Wireless Sensor Networks"/2016 International Conference on Intelligent Sensors /Sensor Networks and Information Processing/ DOI: 10.1109/ISSNIP.2008.4761980

[4] "Trust-Based Decision Making for Environmental Health Community of Interest IoT Systems"2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)
DOI: 10.1109/WiMOB.2016.7763201

[5] "Recommendation-Based Trust Management in Body Location Networks for Mobile Healthcare"2014 IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems. DOI: 10.1109/MASS.2014.85

[6] "Trust-based service management for social internet of things systems"," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 6, pp. 684-696, 2016.

[7] H. A.-H. a. I.-R. Chen*, "Trust-Based Decision Making for Health IoT Systems," *IEEE Internet of Things Journal (Volume: 4 , Issue: 5 , Oct. 2017)*, no. DOI: 10.1109/JIOT.2017.2736446.

[8] "A Trust-based Access Control Scheme for e-Health Cloud," *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA), ,* no. DOI: 10.1109/AICCSA.2018.8612786.

[9] "Trust Chain: Establishing Trust in the IoT-based Applications Ecosystem Using Blockchain"./ IEEE Cloud Computing Co-published by the IEEE CS and IEEE Com Soc July/August 2018.

[10] https://blogs.cisco.com/datacenter/internet-of-things-iot-data-continues-to-explode-exponentially-who-is-using-that-data-and-w